April 2009

UILU-ENG-09-2205
DC-241

A FRAMEWORK FOR SCHEDULING REAL-TIME TRAFFIC OVER WIRELESS CHANNELS

I-Hong Hou and P. R. Kumar

Coordinated Science Laboratory
1308 West Main Street, Urbana, IL 61801
University of Illinois at Urbana-Champaign

REPORT DOCUMENTATION PAGE

Form Approved
OMB NO. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comment regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services. Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	April 2009	

4. TITLE AND SUBTITLE
A Framework for Scheduling Real-Time Traffic over Wireless Channels

5. FUNDING NUMBERS
USARO/W911NF-08-1-0238 and W-911-NF-0710287
NSF/ECCS-0701604, CNS-07-21992, CNS-06-26584, and CSN-05-19535

6. AUTHOR(S)
I-Hong Hou and P. R. Kumar

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
1308 West Main Street
Urbana, Illinois 61801-2307

8. PERFORMING RGANIZATION REPORT NUMBER
UILU-ENG-09-2205
DC-241

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
NSF, 4201 Wilson Blvd, Arlington, VA 22203
USARO, 2800 Powder Mill Rd. Adelphi MD 20783-1197

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official position, policy, or decision, unless so designated by other documentation

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

We develop a general approach for designing scheduling policies for real-time traffic. We extend the model of a previous work, which characterizes a real-time flow by its traffic pattern, delay bound, timely-throughput bound, and channel reliability. The previous work obtained a necessary and sufficient condition for a set of clients to be feasible, and proposed two feasibility optimal scheduling policies. However, the work only considered the case where the deadlines are the same for all clients and the channel state remains static. We extend the model by allowing clients to have different deadlines and by considering time-varying channels. Thus, our model can cope with more realistic fading channels and scenarios with mobile nodes. Based on the extended model, we derive a sufficient condition for a scheduling policy to be feasibility optimal, and thereby a class of feasibility optimal policies. We show that the policies proposed in the previous work both lie in this class, thus generalizing all the results of the previous work. We further demonstrate the utility of the identified class by deriving a feasibility optimal policy for time-varying channels, and suggesting a heuristic for the case where clients have different delay bounds. Simulation results show that our two proposed policies outperform those introduced in the previous work in their respective settings. This result not only shows that the identified class is useful in designing policies under different scenarios, but also suggests that the previous work cannot be applied to more realistic and complicated settings directly.

14. SUBJECT TERMS
Scheduling, wireless channels, traffic pattern

15. NUMBER OF PAGES
12

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

A Framework for Scheduling Real-Time Traffic over Wireless Channels

I-Hong Hou
Department of Computer Science
University of Illinois
Urbana, IL, 61801, USA
ihou2@illinois.edu

P. R. Kumar
CSL and Department of ECE
University of Illinois
Urbana, IL 61801, USA
prkumar@illinois.edu

Abstract

We develop a general approach for designing scheduling policies for real-time traffic. We extend the model of a previous work, which characterizes a real-time flow by its traffic pattern, delay bound, timely-throughput bound, and channel reliability. The previous work obtained a necessary and sufficient condition for a set of clients to be feasible, and proposed two feasibility optimal scheduling policies. However, the work only considered the case where the deadlines are the same for all clients and the channel state remains static. We extend the model by allowing clients to have different deadlines and by considering time-varying channels. Thus, our model can cope with more realistic fading channels and scenarios with mobile nodes. Based on the extended model, we derive a sufficient condition for a scheduling policy to be feasibility optimal, and thereby a class of feasibility optimal policies. We show that the policies proposed in the previous work both lie in this class, thus generalizing all the results of the previous work. We further demonstrate the utility of the identified class by deriving a feasibility optimal policy for time-varying channels, and suggesting a heuristic for the case where clients have different delay bounds. Simulation results show that our two proposed policies outperform those introduced in the previous work in their respective settings. This result not only shows that the identified class is useful in designing policies under different scenarios, but also suggests that the previous work cannot be applied to more realistic and complicated settings directly.

1 Introduction

With the wide deployment of Wireless Local Area Networks (WLANs) and advances in multimedia technology, wireless networks are increasingly being used to carry real-time traffic, such as VoIP and video streaming. These applications usually require certain delay bounds and timely-throughput bounds. In this paper, we study the problem of designing scheduling policies for such applications.

While there has been a lot of research on scheduling real-time traffic over wireline networks, the results are not directly applicable to wireless networks. An important feature of wireless networks is that wireless channels are unreliable and their qualities may be time-varying, either due to fading or node mobility. These features present new challenges to the scheduling problems.

In this work, we consider the scenario where a server is scheduling real-time traffic for a set of clients. We start by studying the results of a previous work [10], which solves the scheduling problem by proposing two feasibility optimal policies in restrictive environments. In particular, the previous work assumes a static channel model, and that all clients in the system require the same delay bound. We extend the model in [10] so that it can capture the traffic patterns, delay bounds, timely-throughput bounds, delivery ratio bounds of clients, for the time-varying wireless channels. Based on this model, we define a Lyapunov function to describe the system behavior. We then establish a sufficient condition for a scheduling policy to be feasibility optimal by analyzing the Lyapunov drift. Based on this we describe a class of policies and prove that they are all feasibility optimal. In general, policies in this class may be computationally complex. We show that, under the more restrictive model in [10], the two proposed policies in the previous work both belong to the identified class. This suggests that the class of policies can serve as a more general guideline to design scheduling policies under different settings.

To further demonstrate the utility of the class of policies, we study two particular cases: one with time-varying channels, and one with clients requiring different delay bounds. For the former case, we derive a scheduling policy and prove that it is feasibility optimal among all priority-based policies. We also obtain a heuristic by studying the class for the case where clients require different delay bounds.

We have also implemented the two derived policies using the IEEE 802.11 standard in a simulation environment. We compare the two derived policies against others, including the policies proposed in the previous work, and a server-centric policy that schedules packets randomly. Simulation results suggest that the two policies outperform others in their corresponding scenarios. In particular, the policies introduced in the previous work fail to offer satisfactory performance. This suggests that neglecting the fact that wireless channels are time-varying, and the possibility that clients may require different delay bounds, can result in significant malperformance of the derived policies.

The rest of the paper is organized as follows: Section 2 reviews some of the related work. Section 3 describes our extension of the model in [10] and derives some useful observations that will be used throughout the paper. In Section 4, we study the results of the previous work under a more restrictive setting. We derive a class of policies that are feasibility optimal in Section 5, which we show to be a generalization of the results in the previous work. Based on this sufficient condition, we obtain a scheduling policy in Section 6, and a heuristic in Section 7. In Section 8, we discuss the implementation issues and demonstrate simulation results. Section 9 concludes this paper.

2 Related Work

Providing QoS over the unreliable wireless channels has received growing interest in recent years. Tassiulas and Ephremides [18] have considered the problem in a single-hop network by assuming ON/OFF channels and derived a throughput-optimal policy. Though the policy is unaware of packet delay, Neely [15] has shown that the average packet delay is constant regardless of the network size under the policy. Andrews et al [1] have proposed another policy that aims to improve packet delay. They have proved that their policy is also throughput optimal but offered no theoretical bound on packet delays. Liu, Wang, and Giannakis [14] have used a cross-layer approach to provide differentiated service for a variety of classes of clients. Grilo, Macedo, and Nunes [8] have proposed a resource-allocation algorithm based on the expected transmission time of each packet. Since the expected transmission time may not be an accurate indication of the actual transmission time, their work cannot provide provable delay guarantees. Raghunathan et al [16] and Shakkottai and Srikant [17] have both approached this problem by analytically demonstrating algorithms to minimize the total number of expired packets in the system. Their results, however, cannot provide differentiated service to different clients. Hou, Borkar, and Kumar [9] have studied the problem of providing QoS based on delay bounds and delivery ratio bounds, and proposed two optimal policies under some restrictive assumptions. Their work has been further extended to deal with variable-bit-rate traffic [10]. In this paper, we extend their work by relaxing some assumptions to deal with more realistic scenarios, such as those with time-varying channels and heterogeneous delay bounds among clients. Fattah and Leung [6] and Cao and Li [3] have surveyed other existing scheduling policies on providing QoS.

3 System Model

In this section, we extend the model proposed in a previous work [10], which only considers a static channel condition and fixed delay bounds for all clients, to account for network behavior and application requirements for providing QoS in wireless systems. Consider a wireless system with N clients, $\{1,2,\ldots,N\}$, and one access point (AP). Packets for clients arrive at the AP, and the AP is in charge of transmitting these packets to respective clients. We assume that time is slotted with normalized slots $t \in \{0,1,2,\ldots\}$, and that time slots are further grouped into *periods* with period T. Packets arrive at the AP at the beginning of each period, at time slots $\{0,T,2T,\ldots\}$, probabilistically, with no more than one packet per client. We model the packet arrivals as a stationary, irreducible Markov process with finite state. The average probability that packets arrive for a subset S of clients is $R(S)$. Notice that we do not assume that the packet arrivals are independent between clients, neither do we assume that the packet arrivals in a period are independent from other periods.

In each time slot, the AP can make exactly one transmission. Each client n specifies a delay requirement τ_n, $\tau_n \leq T$. If the packet for client n is not delivered by the ${\tau_n}^{th}$ time slot of the period, the packet expires and is discarded. This enforces a delay bound of τ_n time slots upon all packets for client n. This scheme naturally applies to a wide range of server-centric wireless communication technologies, such as IEEE 802.11 Point Coordination Function (PCF), WiMax, and Bluetooth.

We consider an unreliable, heterogeneous, and time-varying channel model. We also model the channel condition as a stationary, irreducible Markov process with a finite set of channel states C. The average probability that channel state c occurs is f_c and the channel state remains constant within a period. Under channel state c, the link reliability between the AP and client n is $p_{c,n}$. That is, whenever the AP transmits a packet for client n in a slot, the packet is delivered with probability $p_{c,n}$. The channel state and the packet arrivals in a period are assumed to be independent of each other. We also assume that the AP has knowledge of the current channel state, as well as feedback information on whether a transmission is successful, for example, by requiring the clients to send ACKs upon receiving packets, in which case $p_{c,n}$ is the probability that the AP receives an ACK after making a transmission. Due to the unreliable channels and packet delay bounds, it may be impossible for the AP to deliver every arrived packet. Alternatively, each client n requires a certain timely-throughput bound of q_n packets per period. Since, on average, there are $\sum_{S:n\in S} R(S)$ packets for client n per period, this timely-throughput bound can also be interpreted as a delivery ratio bound of $\frac{q_n}{\sum_{S:n\in S} R(S)}$.

Clients are considered *fulfilled* if the long-term average timely-throughputs meet their respective requirements:

DEFINITION 1. *A set of clients,* $\{1,2,\ldots,N\}$, *with throughput bounds* $[q_n]$, $q_n > 0$ *for all n, is* ***fulfilled*** *under a scheduling policy* η *if for every* $\varepsilon > 0$,

$$Prob\{\frac{d_n(t)}{t/T} > q_n - \varepsilon, \text{ for every } n\} \to 1, \text{ as } t \to \infty,$$

where $d_n(t)$ *is the number of delivered packets for client n up to time t.*

We can capture the current state of the system in a time slot by the current channel state, the set of undelivered packets in the system, and the number of time slots until the next period. Thus, the system behavior can be viewed as a controlled Markov chain. This observation results in the following definition and lemma:

DEFINITION 2. *A* ***stationary randomized policy*** *is one which uses a probability distribution based only on the channel state, the set of undelivered packets, and the number of time slots remaining in the system (and not any events depending on past periods), according to which it randomly chooses an undelivered packet to transmit, or stays idle.*

LEMMA 1. *For any set of clients that can be fulfilled, there exists a stationary randomized policy that fulfills the clients.*

In most work on packet scheduling, the computational overhead is usually assumed to be negligible. However, the computational overhead for some complex policies may be too high for real-time applications. Thus, it may make sense to discuss only a limited set of scheduling policies in some context. We consider one such limited set as the set of *priority-based policies*:

DEFINITION 3. *A* ***priority-based policy*** *is a scheduling policy which assigns priorities to some of the clients, based on past history and current state of the system, at the beginning of each period. During the period, a packet for a client is transmitted only when packets for clients with higher priorities are all delivered. Packets for clients which do not receive a priority are not transmitted. A* ***stationary randomized priority-based policy*** *is one which chooses a priority randomly according to a probability distribution that depends only on the channel state and packet arrivals at the beginning of each period. We denote by* $\mathbb{P}$ *and* $\mathbb{P}_{rand}$ *the set of priority-based policies and the set of stationary randomized priority-based policies, respectively.*

The major advantage of priority-based policies is that all the needed computation is done at the beginning of each period. After priorities are determined, the AP just puts packets into the outgoing queue according to the ordering, and transmits the packet at the head of the queue in every time slot in the period. Thus, priority-based policies are easily implementable.

Based on above definition, we can further define the concept of *feasibility in the set of priority-based policies*:

DEFINITION 4. *A set of clients with timely-throughput bounds* $[q_n]$, $q_n > 0$ *for all n, is* ***feasible in the set*** $\mathbb{P}$ ***of priority-based policies*** *if there exists some scheduling policy in* $\mathbb{P}$ *that fulfills it.*

Analogously to Lemma 1, it can be shown that if $[q_n]$ is feasible in the set $\mathbb{P}$, it is also feasible in the set $\mathbb{P}_{rand}$. Given a set of clients, whether it is feasible in $\mathbb{P}$ is determined by the specifications for the timely-throughput bounds $[q_n]$.

DEFINITION 5. *We call the region in the N-space formed by vectors* $[q_n]$ *for which the clients are feasible in* $\mathbb{P}$, *as the* ***feasibility region under*** $\mathbb{P}$. *Similarly, we can define the* ***feasibility region under the class of all policies.***

In the following lemma, we show that the feasibility region is a convex set.

LEMMA 2. *For any given set of clients, its feasibility region under the class of all policies, as well as the feasible region under* $\mathbb{P}$, *are both convex sets.*

PROOF. Let $[q_n]$ and $[q'_n]$ be two vectors in the feasibility region under $\mathbb{P}$, and thus they are also feasible in $\mathbb{P}_{rand}$. We need to establish that $[\alpha q_n + (1-\alpha)q'_n]$ is also in the feasibility region under $\mathbb{P}$ for all $\alpha \in [0,1]$. Let η and η' be policies in $\mathbb{P}_{rand}$ that fulfill the two vectors, respectively. Since the state of the system at the beginning of a period is not influenced by the enforced scheduling policy, one can design another policy in $\mathbb{P}$ that randomly picks one of the two policies, with η being chosen with probability α, at the beginning of each period. This new policy will fulfill the vector $[\alpha q_n + (1-\alpha)q'_n]$. Further, since q_n and q'_n are both larger than 0 for each n, $\alpha q_n + (1-\alpha)q'_n > 0$ for all n. Thus, the vector $[\alpha q_n + (1-\alpha)q'_n]$ also falls in the feasibility region under $\mathbb{P}$. A similar proof holds for the class of all policies. □

Suppose a set of clients with timely-throughput bounds $[q_n]$ is feasible in $\mathbb{P}$. It is quite obvious that the same set of clients with timely-throughput bounds $[q'_n]$, where $q_n \geq q'_n > 0$ for each n, is also feasible in $\mathbb{P}$. For the ease of discussion, we only consider timely-throughput bounds that are *strictly feasible*:

DEFINITION 6. *A set of clients with timely-throughput bounds* $[q_n]$ *is* ***strictly feasible in*** $\mathbb{P}$ *if there exists some* $\alpha \in (0,1)$ *such that the same set of clients with timely-throughput bounds* $[q_n/\alpha]$ *is feasible in* $\mathbb{P}$.[1]

Finally, we define the concept of *feasibility optimal* policies:

DEFINITION 7. *A scheduling policy* η *is* ***feasibility optimal among*** $\mathbb{P}$ *if it fulfills every set of clients that is strictly feasible in* $\mathbb{P}$.

Similar definitions extend the above terms for the class of all policies. For notational simplicity, in the

[1]Equivalently, $[q_n]$ is strictly feasible if it is an interior point of the feasibility region under $\mathbb{P}$.

rest of the paper, we will not specify the class of policies if we are discussing the one consisting of all policies.

4 Special Case of Static Channel State and Its Generalization

In a previous work [10], Hou and Kumar have studied the problem of admission control and feasibility optimal scheduling for the case where the channel state is static, and all clients require the same delay bounds. (In terms of our model, this means $|C| = 1$ and $\tau_n \equiv \tau$, for all n.) In this section, we briefly introduce their results and show how their results can be generalized for time-varying channel states. We will use p_n, instead of $p_{c,n}$, to represent the channel reliability between the AP and client n, and use τ instead of τ_n since the channel state is static and delay bounds are the same for all clients.

The authors of [10] observed that whether a set of clients is fulfilled is explicitly determined by the portion of time that the AP spends on transmitting packets for each client:

LEMMA 3. *A set of clients is fulfilled if and only if the long-term average number of time slots that the AP spends on transmitting packets for client n per period is at least* $w_n = \frac{q_n}{p_n}$ *for each n.*

Further, since undelivered packets are dropped after the τ^{th} time slot in each period, the number of packets in the system is bounded. Thus, there may be some time slots where the AP may have delivered all packets in the system, and is therefore forced to stay idle. For any subset S of $\{1,2,\dots,N\}$, the authors define I_S to be the minimum number of time slots that the AP is idle in a period for any scheduling policy, given that the AP can only transmit packets for the subset S of clients. Based on these observations, the authors proved a necessary and sufficient condition for strict feasibility:

THEOREM 1. *A set of clients is strictly feasible if and only if* $\sum_{n\in S} w_n < T - E[I_S]$, *for all* $S \subseteq \{1,2\dots,N\}$.

The authors also proposed two scheduling polices and proved they are both feasibility optimal. The two policies are both *largest debt first policies*, where the AP, based on the past history, calculates a debt for each client. In each period, the AP sorts all clients according to their debts and schedules transmissions accordingly, with the packet for client n being scheduled to transmit only after all packets for clients with larger debts have been delivered. In the first policy, the *largest time-based debt first policy*, the debt, which is referred to as the *time-based debt* for client n at time slot t, is defined as $\frac{t}{T}w_n$ minus the number of time slots that the AP has spent on transmitting packets for client n up to time slot t. In the other policy, the *largest weighted-delivery debt first policy*, the so-called *weighted-delivery debt* for client n at time slot t is defined as $\frac{\frac{t}{T}q_n - d_n(t)}{p_n}$, where $d_n(t)$ is the number of delivered packets for client n up to time slot t.

4.1 Extension to Time-Varying Channels

In this section, we discuss how to provide QoS under time-varying channel conditions. One intuitive approach is to decouple the channel states. The AP assigns a timely-throughput bound $q_{c,n}$ for each channel state c and client n, so that the overall timely-throughput for client n is at least q_n; that is, $\sum_{c\in C} f_c q_{c,n} \geq q_n$. Also, for each channel state c, the assigned throughput bounds must be strictly feasible under that channel state, that is, $\sum_{n\in S}\frac{q_{c,n}}{p_{c,n}} < T - E[I_{c,S}]$ for all $S \subseteq \{1,2,\dots,N\}$, where $I_{c,S}$ is the minimal number of time slots that the AP is forced to stay idle in a period under channel state c for any scheduling policy, given that the AP only transmits packets for the subset S of clients. More formally, we therefore seek a matrix $Q = [q_{c,n}]$ that solves the following linear programming problem:

$$\begin{aligned} \text{Max} & \sum_{n=1}^{N}\sum_{c\in C} f_c q_{c,n} \\ \text{s.t.} & \sum_{c\in C} f_c q_{c,n} \geq q_n, \forall n \\ & \sum_{n\in S}\frac{q_{c,n}}{p_{c,n}} \leq T - E[I_{c,S}], \forall c, \forall S \subseteq \{1,2,\cdots,N\}. \end{aligned}$$

After obtaining the matrix Q, we can modify the two largest debt first policies to deal with time-varying channel conditions. Let $s_c(t)$ be the number of time slots up to time slot t that the channel state has been c, and assume that the channel state at time slot t is c. In the largest time-based debt first policy, we define the time-based debt for client n under channel state c as $\frac{s_c(t)}{T}\frac{q_{c,n}}{p_{c,n}}$ minus the number of time slots that the AP has spent on transmitting packets for client n under channel state c up to time slot t. In the largest weighted-delivery debt first policy, we define the weighted-delivery debt for client n under channel state c as $\frac{\frac{s_c(t)}{T}q_{c,n} - d_{c,n}(t)}{p_{c,n}}$, where $d_{c,n}(t)$ is the number of delivered packets for client n under channel state c. Obviously, these two modified largest debt first policies are feasibility optimal.

While this extension offers feasibility optimality, the above linear program involves exponentially many constraints, and solving it is in general computationally inefficient. Further, it also requires the knowledge of the distribution of channel states. In many scenarios, such as those with mobile nodes, this knowledge may not be available. In the following sections, we will describe a more general class of feasibility optimality policies and derive an on-line scheduling policy that is feasibility optimal for the time-varying channel conditions.

5 A Sufficient Condition for Feasibility Optimality

In this section, we describe a more general class of policies that is feasibility optimal. We start by extending the concept of "debt" in the previous work

[10].

DEFINITION 8. *A variable $r_n(k)$, whose value is determined by the past history of the client n up to the k^{th} period, or time slot kT, is called a* ***pseudo-debt*** *if the following properties hold:*

1. *$r_n(0) = 0$, for all n.*
2. *At the beginning of each period, $r_n(k)$ increases by a constant positive number $z_n = z_n(q_n)$, which is an affine function of q_n that increases with q_n.*
3. *At the end of each period, $r_n(k)$ decreases by $\mu_n(k)$, a non-negative and bounded random variable whose value is determined by the behavior of client n during the period.[2] Further, $\mu_n(k) = 0$ if the AP does not transmit any packet for client n during the period.*
4. *The set of clients is fulfilled if and only if $Prob\{\frac{r_n(k)}{k} < \varepsilon\} \to 1$, as $k \to \infty$, for all n and all $\varepsilon > 0$. Alternatively, we can also say that the set of clients is fulfilled if and only if $\frac{r_n(k)^+}{k}$ converges to 0 in probability for all n, where $x^+ := \max\{0, x\}$.*

In the following example, we illustrate that both the time-based debt and the weighted-delivery debt are pseudo-debts under a static channel model.

EXAMPLE 1. *Let $r_n^{(1)}(k)$ denote the time-based debt for client n at time slot kT. It can be interpreted as the following: At the beginning of each period, the time-based debt increases by the amount $w_n = \frac{q_n}{p_n}$. At the end of that period, the time-based debt decreases by the number of time slots that the AP has transmitted packets for client n during the period. Lemma 3 shows that the set of clients is fulfilled if and only if $\frac{r_n^{(1)}(k)^+}{k}$ converges to 0 in probability for all n.*

Similarly, let $r_n^{(2)}(k)$ denote the weighted-delivery debt for client n at time slot kT. The weighted-delivery debt increases by $\frac{q_n}{p_n}$ at the beginning of each period. It decreases by $\frac{1}{p_n}$ if a packet is delivered for client n during that period, and 0 otherwise. By definition, the set of clients is fulfilled if and only if $\frac{r_n^{(2)}(k)^+}{k}$ converges to 0 in probability for all n.

Based on the concept of pseudo-debt, we prove a sufficient condition for feasibility optimality. The proof resembles the one used by Neely [15], though it is used in a different context, and is based on the well-known Lyapunov Drift Theorem:

THEOREM 2 (LYAPUNOV DRIFT). *Let $L(t)$ be a non-negative Lyapunov function. Suppose there exists some constant $B > 0$ and non-negative function $f(t)$ adapted to the past history of the system such that:*

$$E\{L(t+1) - L(t)|\text{history up to time } t\} \le B - \varepsilon f(t),$$

for all t, then: $\limsup_{t\to\infty} \frac{1}{t}\sum_{i=0}^{t} E\{f(i)\} \le B/\varepsilon$. □

We now describe the sufficient condition for feasibility optimality:

THEOREM 3. *Let $r_n(k)$ be a pseudo-debt.*

I. A policy that maximizes the ***payoff function***

$$\sum_{n=1}^{N} E\{r_n(k)^+\mu_n(k)|c_k, S_k, [r_m(k)]\}, \qquad (1)$$

for all $k = 0, 1, 2, \ldots$ is feasibility optimal, where c_k denotes the channel state in the k^{th} period, and S_k is the subset of clients whose packets arrive at the AP in the k^{th} period.

II. If a priority-based policy, i.e., a policy in $\mathbb{P}$, maximizes the payoff function (1) among policies in $\mathbb{P}$, then it is feasibility optimal in $\mathbb{P}$.

PROOF. We present the proof of II. A similar proof works for the class of all policies, too. Define the Lyapunov function: $L(k) = \frac{1}{2}\sum_{n=1}^{N} r_n(k)^2$. Since $r_n(k+1) = r_n(k) + z_n - \mu_n(k)$, the Lyapunov drift can be obtained as follows:

$$\begin{aligned}
&\Delta(L(k)) := E\{L(k+1) - L(k)|[r_m(k)]\}\\
=&E\{\frac{1}{2}\sum_{n=1}^{N} r_n(k+1)^2 - \frac{1}{2}\sum_{n=1}^{N} r_n(k)^2|[r_m(k)]\}\\
=&E\{\sum_{n=1}^{N} r_n(k)[z_n - \mu_n(k)] + \frac{1}{2}\sum_{n=1}^{N}[z_n - \mu_n(k)]^2|[r_m(k)]\}.
\end{aligned}$$

Define $B(k) := E\{\frac{1}{2}\sum_{n=1}^{N}[z_n - \mu_n(k)]^2|[r_m(k)]\}$. $B(k)$ is a bounded random variable and we can assume that $B(k) \le B$, for all k. It follows that for any policy in $\mathbb{P}$:

$$\Delta(L(k)) \le E\{\sum_{n=1}^{N} r_n(k)[z_n - \mu_n(k)]|[r_m(k)]\} + B. \qquad (2)$$

Suppose now that the set of clients, with timely-throughput bounds $[q_n]$, is strictly feasible in $\mathbb{P}$. The vector $[z_n]$ is thus an interior point of the feasibility region (for debt) under $\mathbb{P}$, and there therefore exists some $\alpha \in (0,1)$ such that $[z_n/\alpha]$ is also in the feasibility region under $\mathbb{P}$. Let $z_{min} = \min\{z_1, z_2, \ldots, z_N\}$. The N-dimensional vector $[z_{min}]$ whose elements are all z_{min} falls in the feasibility region under $\mathbb{P}$.Since the feasibility region under $\mathbb{P}$ is a convex set, the vector $\alpha[z_n/\alpha] + (1-\alpha)[z_{min}] = [z_n + (1-\alpha)z_{min}]$ is also in the feasibility region under $\mathbb{P}$.

By Lemma 1, there exists a stationary randomized policy η' in $\mathbb{P}$ that fulfills the set of clients with timely-throughput bounds for the vector $[z_n + (1-\alpha)z_{min}]$. Let $\mu_n'(k)$ be the decrement on the pseudo-debt for client n under η' during the period. Then, we have:

$$\begin{aligned}
E\{\mu_n'(k)|[r_m(k)]\} &= E\{E\{\mu_n'(k)|c_k, S_k, [r_m(k)]\}\}\\
&\ge z_n + (1-\alpha)z_{min},
\end{aligned}$$

for all n. Above, the outer expectation on the right hand side is taken over channel states and the vectors of packet arrivals.

Let η be a policy that maximizes the payoff function (1), for all k, among all policies in $\mathbb{P}$. Then defining $\mu_n(k)$ and $r_n(k)$ as the decrement resulting from

[2] To be exact, $r_n(k+1) = r_n(k) + z_n(q_n) - \mu_n(k)$, for all k.

policy η and the pseudo-debt, we have:

$$\sum_{n=1}^{N} E\{r_n(k)^+\mu_n(k)|c_k, S_k, [r_m(k)]\}$$
$$\geq \sum_{n=1}^{N} E\{r_n(k)^+\mu'_n(k)|c_k, S_k, [r_m(k)]\}.$$

We can assume without loss of generality that the policy does not work on any client n with $r_n(k) \leq 0$, that is, $\mu_n(k) = 0$ if $r_n(k) \leq 0$.[3] Substituting the above inequality into (2), we obtain:

$$\Delta(L(k)) \leq E\{\sum_{n=1}^{N} r_n(k)^+[z_n - \mu_n(k)]|[r_m(k)]\} + B$$
$$\leq E\{\sum_{n=1}^{N} r_n(k)^+[z_n - \mu'_n(k)]|[r_m(k)]\} + B$$
$$\leq -\sum_{n=1}^{N} r_n(k)^+(1-\alpha)z_{min} + B.$$

Let $\varepsilon := (1-\alpha)z_{min}$. By Theorem 2,

$$\limsup_{k\to\infty} \frac{1}{k} \sum_{i=0}^{k} E\{\sum_{n=1}^{N} r_n(k)^+\} \leq B/\varepsilon. \quad (3)$$

Finally, since z_n is a constant and $\mu_n(k)$ is a bounded function, $|r_n(k+1) - r_n(k)|$ is bounded, which implies that $|\sum_{n=1}^{N} r_n(k+1)^+ - \sum_{n=1}^{N} r_n(k)^+|$ is also bounded for all k. Thus, (3) implies that $\frac{1}{k}E\{\sum_{n=1}^{N} r_n(k)^+\} \to 0$ as $k \to \infty$. (See Lemma 4 below). This shows that: $\frac{r_n(k)^+}{k}$ converges to 0 in probability for all n, and η fulfills the set of clients that is strictly feasible in $\mathbb{P}$. Hence, η is feasibility optimal among $\mathbb{P}$. □

It remains to establish the following lemma:

LEMMA 4. *Let $f(t)$ be a non-negative function such that $|f(t+1) - f(t)| \leq M$, for some $M > 0$, for all t. If $\limsup_{t\to\infty} \frac{1}{t}\sum_{i=0}^{t} f(i) \leq B/\varepsilon$, then $\lim_{t\to\infty} \frac{1}{t} f(t) = 0$.*

PROOF. We prove by contradiction. Suppose $\limsup_{t\to\infty} \frac{1}{t} f(t) > \delta$, for some $\delta > 0$. Thus, $f(t) > t\delta$ infinitely often. Suppose $f(t) > t\delta$ for some t. Since $|f(t) - f(t-1)| < M$, we have $f(t-1) > t\delta - M$. Similarly, $f(t-2) > t\delta - 2M$, $f(t-3) > t\delta - 3M, \ldots, f(t - \lfloor t\delta/M \rfloor) > t\delta - \lfloor t\delta/M \rfloor M \geq 0$. Summing over these terms gives us: $\sum_{i=t-\lfloor t\delta/M \rfloor}^{t} f(i) > \frac{t\delta\lfloor t\delta/M \rfloor}{2}$, and thus, $\sum_{i=0}^{t} \frac{1}{t} f(i) > \frac{\delta\lfloor t\delta/M \rfloor}{2}$. Since $f(t) > t\delta$ infinitely often, $\limsup_{t\to\infty} \sum_{i=0}^{t} \frac{1}{t} f(i) = \infty$, which is a contradiction. □

Both of the largest debt first policies proposed in the previous work [10] can be shown to be feasibility optimal under static channel conditions and homogeneous delay bounds, by using Theorem 3. (See Appendix A for detailed proofs). Thus, Theorem 3 offers a more general procedure to design feasibility optimal scheduling policies. To design a scheduling policy in some particular scenarios, we should first choose a proper pseudo-debt and obtain a policy to maximize the payoff function. Maximizing the payoff function is, in general, difficult. However, in some special cases, evaluating the payoff function gives us simple feasibility optimal policies, or, at least, some insights on designing a reasonable heuristic, as long as we choose the correct pseudo-debt. In the following sections, we demonstrate the utility of this approach by deriving policies for the scenarios with both time-varying channels and with heterogeneous delay bounds.

[3] Since a policy cannot lose its feasibility optimality by doing more work, this assumption is not restrictive.

6 Scheduling Policy for Time-Varying Channels

In this section, we propose a scheduling policy for time-varying channels and homogeneous delay bounds. We also show that the policy is feasibility optimal among all priority-based policies.

To derive the scheduling policy, we define $r_n^{(3)}(k) := q_n k - d_n(kT)$, where $d_n(t)$ is the number of delivered packets for client n up to time slot t. We hereafter call $r_n^{(3)}(k)$ as the *delivery debt*. Thus, we have $z_n := q_n$, while $\mu_n(k) = 1$ if a packet for client n is delivered in the period, and $\mu_n(k) = 0$ otherwise.

Suppose at the beginning of some period, the delivery debts are $[r_n^{(3)}(k)]$, the channel state is c, and the set of arrived packets is S. We wish to find the priority ordering that maximizes the payoff function $\mu_{tot}(k) = \sum_{n=1}^{N} r_n^{(3)}(k)^+ E\{\mu_n(k)\}$, where in the expectation we suppose that the channel state is c and the set of arrival packets is S, both fixed. Obviously, transmitting a packet from a client n with $r_n^{(3)}(k) \leq 0$ will not increase the value of $\mu_{tot}(k)$. Thus, we do not give priorities to clients with non-positive delivery debts. This restriction also improves the performance for clients with elastic traffic. In practice, it is possible that clients with real-time traffic and clients with elastic traffic coexist. Thus, in addition to meeting the QoS constraints of those clients with real-time traffic, it is also important not to allocate too much of the resource to those clients and starve those with elastic traffic. For the ease of the remaining discussion, we further assume $r_n^{(3)}(k) > 0$ for all n.

Consider two orderings, A and B: In A, priorities are given as $\{1, 2, \ldots, N\}$, while, in B, priorities are given as $\{1, 2, \ldots, m-1, m+1, m, m+2, m+3, \ldots, N\}$. That is, the second ordering is derived by swapping the orders of clients m and $m+1$ in the first ordering. Let the values of the payoff functions be μ_{tot}^A and μ_{tot}^B for the two orderings. Since clients 1 through $m-1$ have the same priorities in both orderings and their priorities are higher than the remaining clients, the values of $E\{\mu_n(k)\}$, $1 \leq n \leq m-1$ are the same for both orderings. On the other hand, clients $m+2$ through N also have the same priorities in both orderings and they can be scheduled only after the packets for clients 1 through $m+1$ are delivered. The probabilities of packets delivery for these clients are the same under the two orderings. Thus, to compare the two orderings, one only needs to evaluate the probabilities of packets delivery for client m and $m+1$. We further notice that the probabilities that both packets for clients m and $m+1$ are delivered are also the same for both orderings. Let e_n be the event that the packet for client n is delivered, and we have:

$$\begin{aligned}\mu_{tot}^A - \mu_{tot}^B =& r_m^{(3)}(k) Prob\{e_m - e_{m+1} | \text{ordering A}\} \\ & - r_{m+1}^{(3)}(k) Prob\{e_{m+1} - e_m | \text{ordering B}\}.\end{aligned}$$

Suppose that there are τ' time slots left when all packets from client 1 through $m-1$ are delivered. The probability distribution of τ' is the same under both orderings. Recall that the channel reliability for client n under channel state c is $p_{c,n}$. We can further derive:

$$\begin{aligned}&\mu_{tot}^A - \mu_{tot}^B \\ =& r_m^{(3)}(k) E\{\sum_{t=1}^{\tau'} p_{c,m}(1-p_{c,m})^{t-1}(1-p_{c,m+1})^{\tau'-t}\} \\ & - r_{m+1}^{(3)}(k) E\{\sum_{t=1}^{\tau'} p_{c,m+1}(1-p_{c,m+1})^{t-1}(1-p_{c,m})^{\tau'-t}\} \\ =& [r_m^{(3)}(k) p_{c,m} - r_{m+1}^{(3)}(k) p_{c,m+1}] \\ & \times E\{\sum_{t=0}^{\tau'-1} (1-p_{c,m})^t (1-p_{c,m+1})^{\tau'-t-1}\}.\end{aligned}$$

Thus, $\mu_{tot}^A \geq \mu_{tot}^B$ if $r_m^{(3)}(k) p_{c,m} \geq r_{m+1}^{(3)}(k) p_{c,m+1}$. This leads us to obtain the policy described by Algorithm 1 below. Since it jointly considers the delivery debts and the current channel state, we call it the *joint debt-channel policy*. It can be proven to be feasibility optimal among all priority-based policies.

Algorithm 1 Joint Debt-Channel Policy

```
for n = 1 to N do
    r_n^(3)(k) = q_n k − d_n(kT), for all n
Sort clients with a packet arrival such that
    r_1^(3)(k) p_{c,1} ≥ r_2^(3)(k) p_{c,2} ≥ ··· ≥ r_{N_0}^(3)(k) p_{c,N_0} > 0 ≥
    r_{N_0+1}^(3)(k) p_{c,N_0+1} ≥ ...
n ← 1
for each time slot before τ do
    if n ≤ N_0 then
        transmit the packet for client n
        if transmission succeeds then
            n ← n + 1
```

THEOREM 4. *The joint debt-channel policy is feasibility optimal among all priority-based policies.*

PROOF. Let η be the policy described by Algorithm 1, and η' be any priority-based policy. Suppose the priorities assigned by the two policies are $\eta_1, \eta_2, \dots, \eta_m$, and $\eta'_1, \eta'_2, \dots, \eta'_{m'}$, respectively. We modify the priority ordering η' by the following steps:

1. Delete any element in $\eta'_1 \sim \eta'_{m'}$ with $r_{\eta'_n}^{(3)}(k) \leq 0$.
2. For any client n with $r_n^{(3)}(k) > 0$ that is not in $\eta'_1 \sim \eta'_{m'}$, append it at the end of the ordering.
3. If $\eta'_1 \sim \eta'_{m'}$ is still different from $\eta_1 \sim \eta_m$, there exists some n such that $r_{\eta'_n}^{(3)}(k) p_{c,\eta'_n} < r_{\eta'_{n+1}}^{(3)}(k) p_{c,\eta'_{n+1}}$. Swap η'_n and η'_{n+1}.
4. Repeat Step 3 until the two orderings are the same.

Steps 1 and 2 will not decrease the value of the payoff function for η'. As derived above, Step 3 does not decrease the value of the payoff function. Thus, we can conclude that η maximizes the payoff function and is feasibility optimal among all priority-based policies. □

The computation time for Algorithm 1 is only $O(N \log N + \tau)$ in a period, and is more efficient compared to the approach described in Section 4.1. Further, it only requires the information of the current channel state and the debt of each client. In contrast, the approach discussed in Section 4.1 needs the knowledge of the probability distribution of all channel states, which may not always be available. Hence, Algorithm 1 is much easier to implement.

7 A Heuristic for Heterogeneous Delay Bounds

In this section, we describe a heuristic for packet scheduling, for the case where each channel state is static but clients require different delay bounds. We will use p_n to represent the channel reliability between the AP and client n.

We will use the time-based debt, $r_n^{(1)}(k)$, as discussed in Example 1. Thus, we have $z_n := \frac{q_n}{p_n}$, while $\mu_n(k)$ is the number of times the AP transmits the packet for client n in the period. In this case, the payoff function is $E\{\sum_{n=1}^{N} r_n^{(1)}(k)^+ \mu_n(k)\}$.

Suppose, without loss of generality, that at the beginning of a period, packets for clients $\{1, 2, \dots, N_0\}$ arrive at the AP. We further assume that $\tau_1 \leq \tau_2 \leq \dots \leq \tau_{N_0}$. Let γ_n be the number of transmissions the AP needs to make for client n before it can deliver the packet to it. While γ_n is a random variable that cannot be foretold, we ask the following question: how to maximize $\sum_{n=1}^{N_0} r_n^{(1)}(k) \mu_n(k)$ if we *know* the exact values of γ_n?

We resolve this question by proceeding backwards in time. During time slots $[\tau_{N_0-1}+1, \tau_{N_0}]$, all packets except the one for client N_0 has expired, and we can only make transmissions for client N_0 during these time slots. Thus, it does not make sense to schedule client N_0 for more than $\gamma_{N_0}^{N_0-1} := \gamma_{N_0} - (\tau_{N_0} - \tau_{N_0-1})$

transmissions before time slot τ_{N_0-1}. Next we consider the time slots between $[\tau_{N_0-2}+1, \tau_{N_0-1}]$. During these time slots, only clients N_0-1 and N_0 can be scheduled. An obvious choice is to schedule the client with larger debt first, with the restriction that it is not scheduled for more than $\gamma_n^{N_0-1}$ time slots, and to then schedule the other client. (For simplicity, we let $\gamma_{N_0-1}^{N_0-1} := \gamma_{N_0-1}$.) We can further obtain the remaining transmissions allowed for client n before time slot τ_{N_0-2}, which we call $\gamma_n^{N_0-2}$, as $\gamma_n^{N_0-1}$ minus the number of transmissions scheduled for client n during time slots $[\tau_{N_0-2}+1, \tau_{N_0-1}]$. Transmissions of the remaining time slots are scheduled similarly.

Unfortunately, it is impossible to know the exact values of γ_n in advance. Still, we can estimate them. One intuitive way is to estimate them by their expected values, $\frac{1}{p_n}$. However, such estimation does not consider the timely-throughput bounds. If a client has significantly larger debt than others, a reasonably good policy would allocate enough time slots so that the probability of packet delivery for the client in this period is at least its delivery ratio bound, $\frac{q_n}{\sum_{n \in S} R(S)}$, given that client n has a packet arrived in the period. In this work, we estimate γ_n by the number of transmissions that we need to allocate for client n so that it can achieve its delivery ratio bound. Since the channel reliability for client n is p_n, we would therefore estimate γ_n by $\lceil \log_{1-p_n}(1-\frac{q_n}{\sum_{n \in S} R(S)}) \rceil$. We thus derive Algorithm 2 discussed below. Since this heuristic allocates time slots at the beginning of a period, according to the application requirements, channel condition, and system history, we call it the *adaptive-allocation policy*. As in Section 6, we do not schedule transmissions for clients with non-positive debts.

Algorithm 2 Adaptive-Allocation Policy

for $n = 1$ to N **do**
  $r_n^{(1)}(k) =$ time-based debt
  $\gamma_n = \lceil \log_{1-p_n}(1-\frac{q_n}{\sum_{n \in S} R(S)}) \rceil$
Sort clients so that packets for clients $1 \sim N_0$ arrive and $r_1^{(1)}(k) \geq r_2^{(1)}(k) \geq \cdots \geq r_{N_0}^{(1)}(k)$
$alloc \leftarrow n \times 1$−vector
for $t = T$ to 1 **do**
  $n \leftarrow 1$
  while ($\tau_n > t$ or $\gamma_n \leq 0$) and $n \leq N_0$ **do**
    $n \leftarrow n+1$
  if $r_n^{(1)}(k) > 0$ **then**
    $alloc[t] \leftarrow n$
  else
    $alloc[t] \leftarrow N_0+1$
  if $n \leq N_0$ **then**
    $\gamma_n \leftarrow \gamma_n - 1$
for each time slot t **do**
  if $alloc[t] \leq N_0$ and the packet for client $alloc[t]$ has not been delivered **then**
    transmit the packet for client $alloc[t]$
  else
    transmit the packet with the largest positive time-based debt

8 Simulation Results

We have implemented the scheduling policies discussed in previous sections by using the IEEE 802.11 standard in the *ns-2* simulator. In this section, we present the simulation results for the scenario with time-varying channels, and with clients requiring different delay bounds. In each scenario, we compare our policies against the two largest debt first policies in the previous work [10], and a server-centric policy that assigns priorities to clients randomly. Similar to the previous work, we conduct two sets of simulations for each scenario, one with clients carrying VoIP traffic, and one with clients carrying video streaming traffic. The major difference between the two settings lies in their traffic patterns. Many VoIP codecs generate packets periodically. Thus, future packet arrivals can be easily predicted and may be dependent among different clients. For example, if two clients generate packets at the same rate, then either all or none of their packets arrive simultaneously. On the other hand, video streaming technology, such as MPEG, may generate traffic with variable-bit-rate (VBR). Thus, packets arrive at the AP probabilistically, with probability depending on the context of the current frame, and arrivals are independent among different clients. Before showing the simulation results, we first describe the settings of both VoIP traffic and video streaming.

For the VoIP traffic, we follow the standards of the ITU-T G.729.1 [12] and G.711 [11] codecs. Both codecs generate traffic periodically. G.729.1 generates traffic with bit rates 8 – 32 kbits/s, while G.711 generates traffic at a higher rate of 64 kbits/s. We assume the period length, T, is 20 ms, and the payload size of a packet is 160 Bytes. The codecs will generate one packet every several periods; the duration between packet arrivals depends on the bit rate used. We use IEEE 802.11b as the MAC protocol, with data rate 11 Mb/s. Simulation results suggest that the time needed to transmit a packet, including all MAC overheads such as the time for waiting an ACK, is around 610 μs, allowing 32 time slots in a period.

We use MPEG for the video streaming setting. MPEG VBR traffic is usually modeled as a Markov chain consisting of three activity states [13] [5]. Each state generates traffic probabilistically at different mean rates, with the state being determined by the current frame of the video. The statistical mean rates in each state are obtained in an experimental study [5]. We will use the results in setting the traffic patterns of MPEG traffic. Since video streaming requires a much higher bandwidth than VoIP, we use

Table 1: MPEG Traffic Pattern

Activity	Great	High	Regular
Data rate	501597	392237	366587
Arrival probability	1	0.8	0.75

IEEE 802.11a, which can support up to 54 Mb/s data rate, as the underlying MAC. We assume the period length to be 6 ms and the payload size of a packet to be 1500 Bytes. Simulation results show that it takes about 650 μs to transmit a packet and receive the ACK, allowing 9 time slots in a period. Table 1 shows the statistical results by the experimental study [5], where we also present them in terms of the packet arrival probability of our setting. In Table 1, "Data rate" is measured in bits/GoP, where 1 GoP= 240 ms.

For all simulations in this section, we simulate 20 runs for each setting, each run lasting one minute in simulator time. All results shown are averaged over the 20 runs. A natural performance metric for a client is to evaluate the number of more packets the AP needs to deliver for the client to meet its timely-throughput bound, which is the delivery debt, $r_n^{(3)}(k)$, introduced in Section 6. The performance of the system is measured by the sum of positive delivery debts of the clients, that is, $\sum_{n=1}^{N} r_n^{(3)}(k)^+$, which we hereafter call the *total delivery debt*. In addition to evaluating how well the tested policies serve clients with real-time traffic, we also wish to know whether the policies starve those with elastic traffic. Hence, we add a client with saturated elastic traffic in all simulations. Packets for the elastic client are scheduled in all time slots that are left idle by the scheduling policies. We measure the throughput of the client with elastic traffic by the average number of packets it delivers during a simulation.

8.1 Time-varying Channels

In this section, we consider the scenario with time-varying channels, with all clients requiring delay bounds equal to the period length. We model the wireless channel by the widely used Gilbert-Elliot model [4] [7] [19]. In this model, the wireless channel is considered as a two-state Markov chain with a "good" state and a "bad" state. A simulation study by Bhagwat et al [2] shows that the link reliability is 100% when the channel is in the good state, and 20% when the channel is in the bad state. The duration that the channel stays in one state is exponentially distributed with mean 1 – 10 sec for the good state, and 50 – 500 msec for the bad state. We will use this model in our simulation.

While modifying the two largest debt first policies as suggested in Section 4.1 will yield feasibility optimality, such modification requires solving the linear programming problem and is intractable. Rather, we consider some easier modifications for the two policies. For the largest time-based debt first policy, we modify the policy so that it treats the channel as a static one, with link reliability equal to the time-averaged link reliability. For the largest weighted-delivery debt first policy, the weighted-delivery debt for client n at time slot t is defined as $\frac{t}{T}q_n - d_n(t)$ divided by the current link reliability.

For the case with VoIP traffic, we assume there are two groups of clients, A and B. Clients in group A generate one packet every three periods, or at rate 21.3 kbits/s, and requires 90% of each of the clients' packets to be delivered, or a timely-throughput bound of 19.2 kbits/s. Clients in group B generate one packet every two periods at rate 32 kbits/s, and require 70% of each of the clients' packets to be delivered, corresponding to a timely-throughput bound of 22.4 kbits/s. The two groups can be further divided into subgroups, A_1, A_2, A_3, B_1, and B_2. Clients in subgroup A_i generate packets at periods $[i, i+3, i+6, \dots]$, and clients in subgroup B_i generate packets at periods $[i, i+2, i+4, \dots]$. The mean duration of the bad state is 500 msec for all clients, and the mean duration of the good state is $1+0.5n$ sec for the n^{th} client in each subgroup. The time-average link reliability of the n^{th} client in each subgroup can be computed as $\frac{2.2+n}{3+n}$. We assume that there are 14 clients in each of the subgroups.

Simulations results are shown in Figure 1. It can be shown that the joint debt-channel policy incurs near 0 total delivery debt, while all the other policies have much larger total delivery debts. The fact that the largest time-based debt first policy fails to fulfill the set of clients suggests that only considering the average channel reliability without taking channel dynamics into account is not enough for designing scheduling policies. A somewhat surprising result is that the total delivery debt for the largest weighted-delivery debt first policy is even larger than that for the random policy. This is because, by the definition of weighted-delivery debt used in the simulation, the policy, in some sense, favors those clients with poor channels. When the channel state is static, this is not a problem. However, when the channel state is time-varying, it may make more sense to postpone the transmissions for a client with a poor channel until its channel condition turns better. Thus, using weighted-delivery debt for time-varying channels is not only inaccurate, but it is even harmful in some settings. It can also be shown that the throughput for the client with saturated elastic traffic is the highest with the joint debt-channel policy. By only scheduling those real-time clients with positive delivery debts, the policy prevents putting too much effort into any real-time client, and thus reserves enough resources for clients with elastic traffic.

For MPEG traffic, we also assume there are two groups of clients, A and B. Clients in group A generate packets according to Table 1, and clients in group

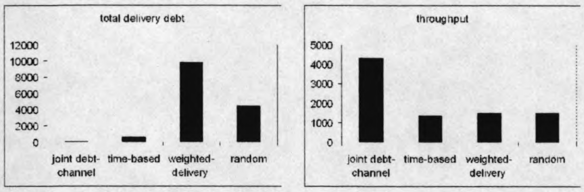

Figure 1: Performance for VoIP traffic under time-varying channels.

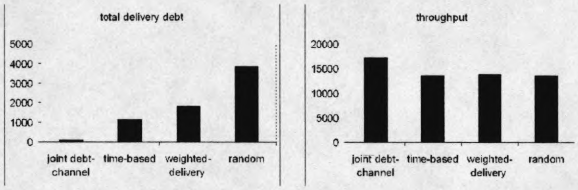

Figure 2: Performance for MPEG traffic under time-varying channels

B are assumed to offer only lower quality video by generating packets only 80% as often as those in group A, in each of the three states. We assume clients in group A require 90% delivery ratios, and clients in group B require 60% delivery ratios. The mean duration when the channel is in the bad state is assumed to be 500 msec for all clients, and the mean duration in the good state is assumed to be $1+0.5n$ sec for the n^{th} client in each group.

Simulation results are shown in Figure 2. As in the case with VoIP traffic, the joint debt-channel policy incurs very small total delivery debt while all the other policies have significantly higher total delivery debts. This result suggests that the simple modifications of the two largest debt first policies do not work under time-varying channels. Also, by only scheduling real-time clients with positive delivery debts, the joint debt-channel policy achieves higher throughput for the client with elastic traffic.

8.2 Heterogeneous Delay Bounds

In this section, we assume that the channel state is static but clients require different delay bounds. Since the length of a period for MPEG traffic is too small, both in terms of time duration (6 ms) and in terms of the number of time slots in a period (9 time slots), it is less meaningful to discuss heterogeneous delay bounds for MPEG traffic. Thus, we only simulate the VoIP case. We assume that there are two groups of clients, A and B. All clients generate traffic at rate 64 kbits/sec, and thus each of them has a packet in each period. Clients in group A require 90% delivery ratios, with delay bounds equal to the period length. On the other hand, clients in group B require 50% delivery ratios, with delay bounds equal to two-thirds of the period length, or 22 time slots. The channel reliability for the n^{th} client in group A is $(84+n)\%$, and that for the n^{th} client in group B is

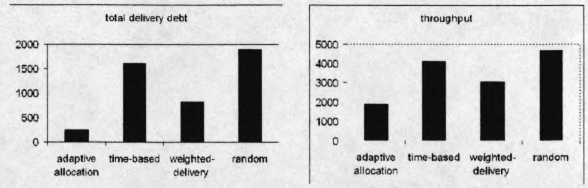

Figure 3: Performance for VoIP traffic under heterogeneous delay bounds

$(29+n)\%$.

Simulation results are shown in Figure 3. The adaptive allocation policy has the smallest total delivery debt among all the tested policies. This is because the other policies, especially the two largest debt first policies, do not consider heterogeneous delay bounds at all. It is not difficult to see that, to maximize the capacity of the system, a policy should, in some sense, work in an "earliest deadline first" fashion. Without considering heterogeneous delay bounds, the largest debt first policies may unwisely schedule clients with longer delay bounds before those with shorter delay bounds, and thus result in poor channel utilization. On the other hand, such poor channel utilization will result in a large number of idle time slots. Thus, the throughputs for the elastic traffic under these policies are higher than those for the adaptive allocation policy.

9 Conclusion

We have analytically studied the problem of scheduling real-time traffic over wireless channels. We have extended a model used in a previous work so as to describe the unreliable wireless channels and real-time application requirements, including traffic patterns, delay bounds, and timely-throughput bounds. Based on the extended model, we have developed a general class of polices that are feasibility optimal. This class can serve as a guideline for designing computational tractable feasibility optimal policy. We have demonstrated the utility of the class by deriving scheduling policies in two special cases, one that deals with time-varying channels, and one that deals with heterogeneous delay bounds. Simulation results have shown that the two policies outperform policies described in the previous work, even though those policies are feasibility optimal in the restrictive environments discussed in the previous work. Thus we have shown not only that the policy class is useful in designing scheduling policies, but also that neglecting some realistic and complicated settings can result in unsatisfactory policies.

10 References

[1] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, P. Whiting, and R. Vijayakumar. Providing quality of service over a shared wireless link. *IEEE Communications Magazine*, 39(2), 2001.

[2] P. Bhagwat, P. Bhattacharya, A. Krishma, and S. K. Tripathi. Using channel state dependent packet scheduling to improve TCP throughput over wireless LANs. *Wireless Networks*, 3(1), 1997.

[3] Y. Cao and V.O.K. Li. Scheduling algorithms in broadband wireless networks. *Proceedings of the IEEE*, 89(1), 2001.

[4] E. O. Elliot. Estimates of error rates for codes on burst-noise channels. *Bell Syst. Tech. J.*, 42, 1963.

[5] I. V. Martin F., J.J. Alins-Delgado, M. Aguilar-Igartua, and J. Mata-Diaz. Modelling an adaptive-rate video-streaming service using Markov-rewards models. In *Proc. of QSHINE*, 2004.

[6] H. Fattah and C. Leung. An overview of scheduling algorithms in wireless multimedia networks. *IEEE Wireless Communications*, 9(5), 2002.

[7] E. N. Gilbert. Capacity of a burst-noise channel. *Bell Syst. Tech. J.*, 39, 1960.

[8] A. Grilo, M. Macedo, and M. Nunes. A scheduling algorithm for QoS support in IEEE802.11 networks. *IEEE Wireless Communications*, 10(3), 2003.

[9] I-H. Hou, V. Borkar, and P.R. Kumar. A theory of QoS for wireless. to appear in *Proc. of IEEE INFOCOM 2009*.

[10] I-H. Hou and P.R. Kumar. Admission control and scheduling for QoS guarantees for variable-bit-rate applications on wireless channels. to appear in *Proc. of ACM MobiHoc 2009*.

[11] ITU-T. Pulse Code Modulation (PCM) of voice frequencies. *ITU-T Recommendations*, 1988.

[12] ITU-T. G.729 based Embedded Variable bit-rate coder: An 8-32 kbit/s scalable wideband coder bitstream interoperable with G.729. *ITU-T Recommendations*, 2006.

[13] L.J. De la Cruz and J. Mata. Performance of dynamic resources allocation with QoS guarantees for MPEG VBR video traffic transmission over ATM networks. In *Proc. of GLOBECOM*, 1999.

[14] Q. Liu, X. Wang, and G.B. Giannakis. A cross-layer scheduling algorithm with QoS support in wireless networks. *IEEE Trans. on Vehicular Technology*, 55(3), 2006.

[15] M. Neely. Delay analysis for max weight opportunistic scheduling in wireless systems. In *Proc. of Allerton Conf.*, 2008.

[16] V. Raghunathan, V. Borkar, M. Cao, and P.R. Kumar. Index policies for real-time multicast scheduling for wireless broadcast systems. In *Proc. of IEEE INFOCOM*, 2008.

[17] S. Shakkottai and R. Srikant. Scheduling real-time traffic with deadlines over a wireless channel. *Wireless Networks*, 8(1), 2002.

[18] L. Tassiulas and A. Ephremides. Dynamic server allocation to parallel queues with randomly varying connectivity. *IEEE Trans. on Information Theory*, 39(2), 1993.

[19] H.S. Wang and N. Moayeri. Finite-state Markov channel – a useful model for radio communication channels. *IEEE Trans. on Vehicular Technology*, 44(1), 1995.

A Another Proof of Feasibility Optimality for Largest Debt First Policies

In this section, we apply Theorem 3 to the two largest debt first policies introduced in [10]. Like in [10], we assume that the channel state is static, with the channel reliability between the AP and client n being p_n, and the delay bounds for all clients being the same, τ. We show that both the largest time-based debt first policy and the largest weighted-delivery debt first policy are feasibility optimal (among all policies).

THEOREM 5. *Let $r_n^{(1)}(k)$ be the time-based debt as defined in Example 1. Then, the largest time-based debt first policy maximizes the payoff function:*

$$\sum_{n=1}^{N} E\{r_n^{(1)}(k)^+\mu_n(k)|S_k,[r_m^{(1)}(k)]\},$$

where $\mu_n(k)$ is the number of transmissions the AP makes for client n during the k^{th} period, and S_k is the subset of clients that have a packet during the period.

PROOF. Without loss of generality, assume that clients $1,\dots,N_0$ have a packet during the period, and $r_1^{(1)}(k)\geq r_2^{(1)}(k)\geq\cdots\geq r_{N_0}^{(1)}(k)$. Let γ_n, $1\leq n\leq N_0$, be the random variable denoting the number of transmissions the AP has to make for client n before a successful transmission. Recall that μ_n is the number of transmissions that the AP actually makes for client n during the period. Assuming that $[\gamma_n]$ is known, maximizing the payoff function reduces to solving the following linear programming problem:

$$\text{Max} \sum_{n=1}^{N_0} r_n^{(1)}(k)^+\mu_n \qquad \text{s.t. } \mu_n\leq\gamma_n, \forall n$$
$$\sum_{n=1}^{N_0}\mu_n\leq\tau.$$

One obvious solution is to allocate the first γ_1 time slots to client 1, the next γ_2 time slots to client 2, etc., until all the first τ time slots are allocated or all packets are delivered. This solution is consistent with the largest time-based debt first policy. That is, the largest time-based debt first policy maximizes $\sum_{n=1}^{N_0} r_n^{(1)}(k)^+\mu_n$ for every sample path, and thus maximizes the payoff function. $\square$

THEOREM 6. *The largest weighted-delivery debt first policy is feasibility optimal.*

PROOF. Let $r_n^{(2)}(k)$ be the weighted-delivery debt as defined in Example 1. We first show that the largest weighted-delivery debt first policy maximizes the payoff function, $\sum_{n=1}^{N} E\{r_n^{(2)}(k)^+\mu_n(k)|S_k,[r_m^{(2)}(k)]\}$, among all priority-based policies, where $\mu_n(k)=\frac{1}{p_n}$ if the packet for client n is delivered in the period, and 0 otherwise.

The proof is very similar to the argument used in Section 6. Suppose clients $1,\dots,N_0$ have a packet in the period. Let A be the ordering $\{1,2,\dots,m,m+1,\dots,N_0\}$, and B the ordering $\{1,2,\dots,m-1,m+1,m,\dots,N_0\}$. Let μ_{tot}^A and μ_{tot}^B be the values of payoff functions for the two orderings, and e_n the event that the packet for client n is delivered. Similar to Section 6, we have:

$$\mu_{tot}^A-\mu_{tot}^B = r_m^{(2)}(k)^+Prob\{e_m-e_{m+1}|\text{A}\}/p_m - r_{m+1}^{(2)}(k)^+Prob\{e_{m+1}-e_m|\text{B}\}/p_{m+1}.$$

Suppose there are τ' time slots left before packets expire when all packets from clients $1,\ldots,m-1$ are delivered. Then $Prob\{e_m - e_{m+1}|\mathrm{A}\} = \sum_{t=1}^{\tau'} p_m(1-p_m)^{t-1}(1-p_{m+1})^{\tau'-t}$, and $Prob\{e_{m+1} - e_m|\mathrm{B}\} = \sum_{t=1}^{\tau'} p_{m+1}(1-p_{m+1})^{t-1}(1-p_m)^{\tau'-t}$. Thus we have:

$$\begin{aligned}\mu_{tot}^A - \mu_{tot}^B =& (r_m^{(2)}(k)^+ - r_{m+1}^{(2)}(k+1)^+) \\ & \times E\{\sum_{t=1}^{\tau'}(1-p_{m+1})^{t-1}(1-p_m)^{\tau'-t}\},\end{aligned}$$

and $\mu_{tot}^A \leq \mu_{tot}^B$ if $r_m^{(2)}(k) \leq r_{m+1}^{(2)}(k)$.

Now, let $\eta_1, \eta_2, \ldots$ be any other ordering assigned by some priority-based policy. We modify it into one assigned by the largest weighted-delivery debt first policy in the following steps:

1. For any client with a packet that is not assigned a priority, append it at the end of the ordering.
2. If the resulting ordering is different from the one obtained by the largest weighted-delivery debt first policy, there exists some m such that $r_{\eta_m}^{(2)}(k) < r_{\eta_{m+1}}^{(2)}(k)$. Swap the order of η_m and η_{m+1}.
3. Repeat Step 2 until the ordering is the same as the one by the largest weighted-delivery debt first policy.

Obviously, Step 1 cannot decrease the value of the payoff function. By the argument in the previous paragraphs, Step 2 cannot decrease the value of the payoff function either. Thus, the value of the payoff function of the largest weighted-delivery debt first policy is no smaller than that of any other priority-based policy. This implies that this policy is feasibility optimal among all priority-based policies. Further, by Theorem 5, there exists a priority-based policy, namely, the largest time-based debt first policy, that is feasibility optimal among all policies and hence so is the largest weighted-delivery debt first policy. □